

J1036 U.S. PTO
10/015768
12/17/01

대한민국특허청
KOREAN INTELLECTUAL
PROPERTY OFFICE

별첨 사본은 아래 출원의 원본과 동일함을 증명함.

This is to certify that the following application annexed hereto is a true copy from the records of the Korean Intellectual Property Office.

출원번호 : 특허출원 2001년 제 65337 호
Application Number PATENT-2001-0065337

출원년월일 : 2001년 10월 23일
Date of Application OCT 23, 2001

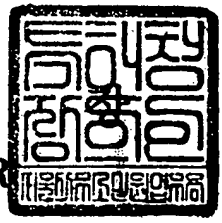
출원인 : 한국전자통신연구원
Applicant(s) KOREA ELECTRONICS & TELECOMMUNICATIONS RESEARCH INST



2001 년 11 월 22 일

특 허 청

COMMISSIONER



CERTIFIED COPY OF
PRIORITY DOCUMENT

【서지사항】

| | |
|------------|--|
| 【서류명】 | 특허출원서 |
| 【권리구분】 | 특허 |
| 【수신처】 | 특허청장 |
| 【참조번호】 | 0003 |
| 【제출일자】 | 2001.10.23 |
| 【발명의 명칭】 | 클러스터 컴퓨터 시스템의 소프트웨어 가용도 개선 방법 및 그 장치 |
| 【발명의 영문명칭】 | Apparatus and method for improving the availability of cluster computer systems |
| 【출원인】 | |
| 【명칭】 | 한국전자통신연구원 |
| 【출원인코드】 | 3-1998-007763-8 |
| 【대리인】 | |
| 【성명】 | 권태복 |
| 【대리인코드】 | 9-2001-000347-1 |
| 【포괄위임등록번호】 | 2001-057650-1 |
| 【대리인】 | |
| 【성명】 | 이화익 |
| 【대리인코드】 | 9-1998-000417-9 |
| 【포괄위임등록번호】 | 1999-021997-1 |
| 【발명자】 | |
| 【성명의 국문표기】 | 박기진 |
| 【성명의 영문표기】 | PARK,Kie-Jin |
| 【주민등록번호】 | 660705-1023219 |
| 【우편번호】 | 305-340 |
| 【주소】 | 대전광역시 유성구 도룡동 431-6 현대아파트 102동 904호 |
| 【국적】 | KR |
| 【발명자】 | |
| 【성명의 국문표기】 | 김성수 |
| 【성명의 영문표기】 | KIM,Sung-Soo |
| 【주민등록번호】 | 600211-1037215 |
| 【우편번호】 | 442-380 |

【주소】 경기도 수원시 팔달구 원천동 산 5번지 아주대학교
정보통신대학원
【국적】 KR
【발명자】
【성명의 국문표기】 김상현
【성명의 영문표기】 KIM, Sang-Hyun
【주민등록번호】 630215-1452453
【우편번호】 302-283
【주소】 대전광역시 서구 월평3동 누리아파트 103동 603호
【국적】 KR
【발명자】
【성명의 국문표기】 김장경
【성명의 영문표기】 KIM, Jang-Kyung
【주민등록번호】 571116-1177414
【우편번호】 302-741
【주소】 대전광역시 서구 만년동 강변아파트 112동 802호
【국적】 KR
【발명자】
【성명의 국문표기】 박중무
【성명의 영문표기】 PARK, Joong-Moo
【주민등록번호】 460211-1058312
【우편번호】 305-345
【주소】 대전광역시 유성구 신성동 한울아파트 109동 1101호
【국적】 KR
【심사청구】 청구
【취지】 특허법 제42조의 규정에 의하여 위와 같이 출원합니다. 대리인
권태복 (인) 대리인
이화익 (인)
【수수료】
【기본출원료】 20 면 29,000 원
【가산출원료】 24 면 24,000 원
【우선권주장료】 0 건 0 원
【심사청구료】 14 항 557,000 원

1020010065337

출력 일자: 2001/11/23

| | |
|-----------|-------------------|
| 【합계】 | 610,000 원 |
| 【감면사유】 | 정부출연연구기관 |
| 【감면후 수수료】 | 305,000 원 |
| 【첨부서류】 | 1. 요약서·명세서(도면)_1통 |

【요약서】**【요약】**

본 발명은, 여러 대의 서버로 구성된 클러스터 컴퓨터 시스템 관리자가 예측할 수 있는 적절한 시점을 택하여 프로그램을 일시적으로 중지시킨 후, 다시 가동시키는 소프트웨어 재할 기법을 이용한 클러스터 컴퓨터 시스템의 소프트웨어 가용도 개선 방법 및 그 장치에 관한 것으로, 소프트웨어 측면과 하드웨어 측면을 모두 고려하였으며, 소프트웨어 재할에 의한 예방적 차원의 결함 허용 기법을 활용한 것이고, 클러스터 시스템의 소프트웨어적인 불안정률과 하드웨어적인 고장률에 따른 최적 재할 시기의 결정을 통하여 가용도를 개선함으로써, 비용 측면에 있어서 효율적인 고 가용성 컴퓨터 시스템의 특성을 확보할 수 있도록 한 것이다.

【대표도】

도 9

【색인어】

클러스터, 주 서버, 여분서버, 재할, 절체, 메모리, 가용서버, 가동모드

【명세서】

【발명의 명칭】

클러스터 컴퓨터 시스템의 소프트웨어 가용도 개선 방법 및 그 장치
{Apparatus and method for improving the availability of cluster computer systems}

【도면의 간단한 설명】

도 1은 일반적인 클러스터 컴퓨터 시스템의 블록 구성을 나타낸 도면.

도 2는 종래 기술에 따른 클러스터 컴퓨터 시스템의 상태 전이 모델을 나타낸 도면.

도 3은 본 발명에 따른 소프트웨어 재활을 고려한 클러스터 컴퓨터 시스템의 상태 전이 모델을 나타낸 도면.

도 4는 본 발명에 따른 이중화된 클러스터 시스템의 소프트웨어 재활 기법 적용 예시도.

도 5는 본 발명에 따른 소프트웨어 가용도 개선 장치를 포함하는 클러스터 컴퓨터 시스템의 구성을 나타낸 도면.

도 6은 도 5에 도시된 클러스터링 모듈의 상세 구성을 나타낸 도면.

도 7은 도 5에 도시된 소프트웨어 재활 모듈의 상세 구성을 나타낸 도면.

도 8은 도 5에 도시된 결합 허용 모듈의 상세 구성을 나타낸 도면.

도 9는 도 6, 7, 8에 도시된 본 발명에 따른 클러스터 컴퓨터 시스템의 소프트웨어 가용도 개선 장치의 상호 연결 구성을 나타낸 도면.

도 10은 본 발명에 따른 클러스터 컴퓨터 시스템의 소프트웨어 가용도 개선 방법에 있어서, 서버의 불안정한 상태(소프트웨어의 불안정 상태) 발생시 이를 복구하는 방법에 대한 동작 플로우차트를 나타낸 도면.

도 11은 본 발명에 따른 클러스터 컴퓨터 시스템의 소프트웨어 가용도 개선 방법에 있어서, 서버의 결함(하드웨어 결함으로 서비스 중단시)시 이를 복구하는 방법에 대한 동작 플로우차트를 나타낸 도면.

도면의 주요 부분에 대한 부호의 설명

500 : 고 가용성 클러스터 컴퓨터 시스템

501 : 클러스터링 모듈 502 : 소프트웨어 재활 모듈

503 : 결함 허용 모듈 504 : 주 서버

505 : 여분 서버 601 : 이중화 구조의 부하 분배기

602 : 클러스터 제어기 701 : 재활 명령 생성기

702 : 시스템 상태 수집기 703 : 시스템 감시기

801 : 결함 감지기 802 : 결함 복구기

803 : 결함 절체기

【발명의 상세한 설명】

【발명의 목적】

【발명이 속하는 기술분야 및 그 분야의 종래기술】

<21> 본 발명은 클러스터 컴퓨터 시스템의 소프트웨어 가용도 개선 방법 및 그 장치에 관한 것으로, 특히 여러 대의 서버로 구성된 클러스터 컴퓨터 시스템의 결함이 발생하기 전에 이를 미연에 방지하는 능동적 차원의 결함 허용 방법이다. 즉, 본 발명은 클러스터 컴퓨터 시스템의 가용도를 개선하기 위하여 불시의 소프트웨어 장애로 인한 피해가 없이 컴퓨터 시스템 관리자가 예측할 수 있는 적절한 시점(예를 들면, 시스템의 이용이 한산하다든지, 프로그램이 문제를 야기시킬 우려가 있다든지)을 택하여 프로그램을 일시적으로 중지시킨 후, 다시 가동하는 소프트웨어 재활 기법을 사용한 클러스터 컴퓨터 시스템의 소프트웨어 가용도 개선 방법 및 그 장치에 관한 것이다.

<22> 일반적으로, 컴퓨터 시스템 기술의 발전과 사용자의 다양한 요구로 인하여 소프트웨어의 복잡도는 날로 증가되는 추세에 있으며, 이에 따라 소프트웨어의 설계, 구현 또는 그 밖의 여러가지 원인과 소프트웨어와 관련된 결함으로 인하여 전체 시스템의 오동작 또는 수행 중단으로 이어지는 사례가 늘어나고 있다.

<23> 컴퓨터 장애의 원인으로는 하드웨어/소프트웨어 결함에 의한 장애가 있으며, 조사된 연구에 따르면 소프트웨어에 의한 장애는 그 비중이 점차로 늘어날 전망이다. 소프트웨어의 결함을 줄이기 위한 방법으로 소프트웨어의 결함을 설계,

구현, 시험 과정에서 발견하여 이를 해결하는 방법이 가장 이상적이지만 현실적으로 결함이 전무한 상태로 만드는 것은 불가능하다.

<24> 소프트웨어 재활 기법은 점차로 복잡해져 가는 인터넷, 실시간 처리 등의 클라이언트-서버(Client-Server) 컴퓨팅 환경에 적합한 대안이라 평가되고 있고, 버퍼 플러싱, 메모리 청소, 파일 시스템 정리, 커널 테이블의 초기화 등을 재시동 방법으로 처리할 수 있으며, 이 경우 소프트웨어는 일시적 결함(Transient Fault)이 발생할 확률이 작은 새로운 상태에서 재출발할 수 있게 된다.

<25> 기존의 소프트웨어 결함 허용 방식(예를 들면, Recovery Block, N-version Programming, N-self Checking Programming, Checkpointing)들은 새로운 컴퓨팅 환경 변화에 대한 적응력이 떨어지며, 네트워크 환경과 응용 프로그램의 복잡, 대형화 추세로 인해 대규모 실시간 트랜잭션 처리 응용 프로그램(인터넷 검색 서비스, बैं킹, 증권 전산 등)들의 고 가용성 확보 방안으로는 적합하지 않다.

<26> 클라이언트-서버 형태의 컴퓨팅 환경의 서버에서 가동되는 소프트웨어는 상당히 긴 시간동안 수행 가능하여야 하며, 서버 소프트웨어의 가동 시간이 길어질수록 다수의 클라이언트의 요청으로 인한 에러 데이터의 누적이 필연적이다. 장시간 가동으로 인한 소프트웨어 노화(Software Aging)로 인해서 시스템의 성능 저하 및 일시적 결함의 발생 가능성이 커지게 되며, 결국에는 메모리 부족, 화일 공유의 오류, 또는 데이터 손상 등과 같은 소프트한 결함으로 인해서 서버 소프트웨어의 성능이 점차로 저하되어 이것이 지속될 경우 시스템의 작동이 멈추게 된다.

- <27> 소프트웨어 노화로 인한 결함(결함 허용 분야에서 이와 같은 유형의 결함을 Heisenbugs라고 함)의 감지, 수정은 상당히 어려우며, 단순히 하드웨어적으로 시스템의 가용도를 높이기 보다는 소프트웨어적인 결함으로 인한 시스템 장애를 사전에 예방함으로써, 서비스가 중단되는 평균 횟수를 최소화하는 것이 더 바람직하다.
- <28> 하드웨어 기술의 발전으로 인하여 시스템 가용도 척도에 하드웨어 보다는 소프트웨어가 더욱 더 큰 영향을 끼치고 있으며, 특히 복잡한 대규모 소프트웨어의 등장으로 결함이 없는 소프트웨어의 개발은 거의 불가능하게 되므로 인해 소프트웨어 결함 허용에 대한 필요성이 점차 중요시 되고 있다. 대부분의 소프트웨어 결함이 영구적이기 보다는 일시적 결함의 성질을 가지고 있으며, 소프트웨어 노화로 인해 발생하는 일시적 결함은 시스템을 재 가동할 경우에 대부분 없어진다.
- <29> 도 1은 일반적인 클러스터 컴퓨터 시스템의 블록 구성을 나타낸 도면으로서, 도 1에 따르면, 클라이언트와 서버는 ADSL(Asymmetric Digital Subscriber Line), 이더넷(Ethernet), 케이블, 랜(LAN)등과 같은 고속 가입자 망을 통해 연결되며, 서버의 데이터는 SCSI(Small Computer System Interface)나, 광 채널 인터페이스, TCP-IP등을 통해 하드 디스크 등의 저장장치(도면에서는 다수의 디스크 어레이로 표시됨)에 의해 관리된다.
- <30> 이와 같은 일반적인 클러스터 컴퓨터 시스템에 대한 종래 기술에 따른 상태전이 모델에 대하여 도 2를 참조하여 살펴보자.

<31> 도 2는 장시간 가동된 소프트웨어의 불안정성을 고려하지 않은 종래 기술에 따른 클러스터 컴퓨터 시스템의 상태 전이 모델을 나타낸 도면이다.

<32> 도 2에서 소프트웨어 재활을 고려한 가용도(Availability)는 모든 서버가 고장이거나(P_0), 한대의 서버가 가동중일 경우 재활 작업을 수행하는(P_{r1})상태 확률을 제외한 경우이기 때문에 아래의 수학적 식 1과 같이 표현될 수 있다.

<33>
$$\text{Availability} = 1 - (P_0 + P_{r1})$$
 【수학적 식 1】

<34> 불시에 발생한 결함 혹은 소프트웨어 재활 작업으로 인해 서비스를 제공할 수 없는 다운타임(Downtime)은 클러스터 컴퓨터 시스템의 가동 시간(T)에 대한 함수로 아래의 수학적 식 2와 같이 표현될 수 있다.

<35>
$$\text{Downtime}(T) = (1 - \text{Availability}) * T$$
 【수학적 식 2】

<36> 서버의 가동 정지로 인한 단위 시간당 손실 비용을 C_f , 소프트웨어 재활 작업으로 인한 단위 시간당 손실 비용을 C_r 이라 할 경우 서버의 정지로 인해 발생하는 비용은 아래의 수학적 식 3과 같다.

<37>
$$\text{Cost}(T) = (P_0 * C_f + P_{r1} * C_r) * T$$
 【수학적 식 3】

<38> 일반적으로 예상 가능한 시스템 정지 비용은 불시 정지로 인한 손실 비용에 비해 훨씬 저렴하게 된다($C_f \gg C_r$).

<39> 클러스터 컴퓨터 시스템에서 수행되는 소프트웨어의 재활 주기, 재활 소요 시간, 서버의 하드웨어적인 고장률과 수리율, 동시에 가동되는 서버의 수 및 서버의 가동 기간 등의 시스템 운영 파라미터에 기초하여 소프트웨어 재활학적으로

검증함으로써, 소프트웨어 재활을 통한 예방적 결함 허용 기법의 적용 가능성이 높다는 것을 확인하였다.

<40> 또한, 장시간 가동으로 인한 소프트웨어적인 불안정률 및 서버의 하드웨어적인 고장률이 클러스터 시스템의 가용도 개선에 중요한 특성 요소임을 파악하였다.

<41> 그러나, 이와 같은 종래 기술에 따른 컴퓨터 시스템의 가용도 개선을 위한 소프트웨어 재활 기법에서는 고가의 이중화된 대용량 서버 시스템을 대상으로 하고 있으며, 최근 각광받고 있는 비용 효율적인 고성능, 고 가용성 솔루션인 클러스터 컴퓨터 시스템을 대상으로 하고 있지 않다. 따라서, 비용 효율적인 고 가용성 시스템의 구축이 어려운 문제점이 야기된다.

【발명이 이루고자 하는 기술적 과제】

<42> 따라서, 본 발명은 상기한 종래 기술에 따른 제반 문제점을 해결하기 위하여 안출한 것으로, 본 발명의 목적은, 여러 대의 서버로 구성된 클러스터 컴퓨터 시스템 관리자가 예측할 수 있는 적절한 시점을 택하여 프로그램을 일시적으로 중지시킨 후, 다시 가동시키는 소프트웨어 재활 기법을 이용한 클러스터 컴퓨터 시스템의 소프트웨어 가용도 개선 방법 및 그 장치를 제공함에 있다. 즉, 소프트웨어 측면과 하드웨어 측면을 모두 고려하였으며, 소프트웨어 재활에 의한 예방적 차원의 결함 허용 기법을 활용한 클러스터 컴퓨터 시스템의 소프트웨어 고 가용도 개선 방법 및 그 장치를 제공하는 것이다.

<43> 또한, 본 발명의 다른 목적은, 클러스터 시스템의 소프트웨어적인 불안정률과 하드웨어적인 고장률에 따른 최적 재할 시기의 결정을 통하여 가용도를 개선함으로써, 비용 효율적인 고 가용성 컴퓨터 시스템의 특성을 확보할 수 있도록 한 클러스터 컴퓨터 시스템의 소프트웨어 가용도 개선 방법 및 그 장치를 제공함에 있다.

【발명의 구성 및 작용】

<44> 상기한 목적을 달성하기 위한 본 발명에 따르면, 클러스터를 구성하는 서버 하드웨어 고장률, 서버에 탑재된 소프트웨어의 장시간 가동으로 인한 불안정 상태를 반영하는 불안정률, 결함 발생 가능성이 작은 시스템 초기 가동 상태로 되기 위해 필요한 재할 소요 시간, 클러스터 시스템의 연속 가동 기간 및 단위 시간 당 손실 비용 등의 파라미터로부터 가용도를 최대화하고 손실 비용을 최소화해주는 최적 재할 기법을 제시하는 방식으로 고가용도를 발생한 것이다.

<45> 본 발명에 따른 클러스터 컴퓨터 시스템의 소프트웨어 가용도 개선 방법의 일측면에 따르면, 상기 다수의 주 서버에 대한 시스템 상태 정보를 수집하여 상기 서버들의 이상 상태 유무를 감시하는 단계; 상기 감시결과, 임의의 서버가 불안정한 상태로 판단되면, 여분 서버 또는 여분 용량이 존재하는 다른 주 서버가 존재하는지를 판단하는 단계; 여분 서버 또는 여분 용량을 가진 임의의 주 서버가 존재하는 경우, 현재 설정되어 있는 가동 모드에 따라 불안정한 상태의 주 서버를 여분 서버 또는 여분 용량이 있는 다른 주 서버로 모든 프로세스를 이중화하는 단계; 이중화가 완료되면, 불안정한 상태로 판단된 해당 서버에 재할 동작

을 수행하기 위한 시스템 재할 제어 신호를 제공하는 단계를 포함한다. 여기서, 상기 시스템 상태 정보는, 주 서버의 작업 부하, 연속 가동 시간, 메모리 사용량, 버퍼 사용량 정보 중 적어도 하나의 정보를 포함한다.

<46> 또한, 상기 모든 프로세스를 이중화하는 단계는, 상기 현재 설정된 모드가 액티브/스탠바이(Active/Standby) 또는 액티브/액티브(Active/Active) 가동 모드인 경우, 임의의 여분 서버 또는 여분 용량을 가지는 주 서버를 선정하는 단계; 상기 불안정한 상태에 있는 주 서버의 모든 프로세스를 상기 선정된 여분 서버에 이중화시키고, 상기 불안정한 상태에 있는 주 서버의 모든 프로세스를 상기 선정된 여분 용량을 가진 주 서버에 이중화시키는 단계를 포함한다.

<47> 또한, 상기 서버의 재할 동작을 수행하는 단계는, 재할 대상이 되는 주 서버의 이중화가 완료되면, 재할 대상이 되는 주 서버의 작업 부하, 연속 가동 시간에 따라 재할 명령 수행 여부를 판단하는 단계; 상기 판단 결과 재할 명령 수행이 이루어져야 하는 경우, 가용 서버 리스트에 상기 재할 대상이 되는 주 서버의 리스트를 삭제하는 단계; 상기 이중화된 여분 서버를 주 서버로 전환한 후, 상기 재할 대상이 되는 주 서버의 재할 동작을 수행하는 단계; 재할 동작이 완료되면, 재할 동작이 완료된 주 서버를 가용 서버 리스트에 여분 서버로 등록하는 단계를 포함할 수 있다. 여기서, 상기 재할 대상이 되는 주 서버의 재할 동작은, 파일 시스템 청소, 버퍼 청소, 메모리 청소, 재 부팅 동작을 포함한다.

<48> 또한, 본 발명에 따른 클러스터 컴퓨터 시스템의 결함 감시 방법의 일 측면에 따르면, 상기 다수의 주 서버 각각에 대해 고장으로 인한 서비스의 중단을 감지하는 단계; 상기 감지결과, 임의의 주 서버에 결함이 발생되어 서비스가 중단

되면, 주 서버를 임의의 부 서버로 절체한 후, 결함이 발생된 주 서버의 결함 복구 명령을 발생하는 단계; a) 상기 발생된 복구 명령에 따라 주 서버에 대한 모든 기능을 상기 부 서버로 전이하고, b) 여분 서버로의 전이가 완료되면, 여분 서버를 주 서버로 등록하고, 결함이 발생된 상기 주 서버를 가용 서버 리스트로부터 삭제하는 단계; 상기 가용 서버 리스트로부터 삭제된 서버의 결함을 복구한 후, 결함이 복구된 서버를 부 서버로 가용 서버 리스트에 등록하는 단계를 포함한다.

<49> 한편, 본 발명에 따른 클러스터 컴퓨터 시스템의 소프트웨어 가용도 개선 장치의 일 측면에 따르면, 상기 다수의 주 서버에 대한 시스템 상태 정보를 수집하여 각 서버의 불안정한 상태를 파악하는 시스템 감시 수단; a) 상기 시스템 감시 수단에서 시스템 감시 결과, 임의의 주 서버가 불안정한 상태인 경우, 현재 설정되어 있는 가동 모드에 따라 불안정한 상태의 주 서버를 여분 서버 또는 여분 용량이 있는 다른 주 서버로 모든 프로세스를 이중화시키기 위한 이중화 제어 신호를 제공하고, b) 상기 불안정한 상태에 있는 주 서버의 시스템 상태가 일정 시간 동안 지속되는 경우 해당 주 서버에 시스템 재활을 위한 재활 신호를 제공하는 클러스터 제어 수단; 상기 클러스터 제어 수단에서 제공되는 설정된 모드에 대한 이중화 제어 신호에 따라 불안정한 상태의 주 서버의 모든 프로세스를 여분 서버 또는 여분 용량이 존재하는 다른 주 서버로 이중화시키는 이중화 수단을 포함한다.

- <50> 또한, 상기 시스템 감시 수단은, 상기 주 서버들의 시스템 상태를 감시하여 각 서버들의 상태 정보를 수집하는 시스템 상태 정보 수집부; 상기 시스템 상태 정보 수집부에서 수집된 시스템 상태 정보에 따라 불안정한 주 서버가 존재하는지를 판단하고, 불안정한 상태의 주 서버가 존재하는 경우 해당 주 서버의 불안정한 소프트웨어의 재활을 위한 재활 명령 신호를 생성하여 상기 이중화 수단으로 제공하는 재활 명령 생성부를 포함한다.
- <51> 또한, 상기 클러스터 제어수단은, 상기 이중화 수단에서 상기 불안정한 상태의 주 서버가 여분 서버 또는 여분 용량을 가진 다른 주 서버로 이중화된 경우 상기 불안정 상태의 주 서버를 가용 서버 리스트로부터 삭제시키고, 상기 재활 신호에 따라 불안정한 상태의 주 서버의 재활이 완료되면 재활이 완료된 주 서버를 가용 서버 리스트로 재 등록하는 등록부를 포함한다.
- <52> 또한, 상기 이중화 수단은, 상기 클러스터 제어 수단에 설정된 가동 모드에 따라 여분 서버 또는 여분 용량을 가진 임의의 주 서버를 선정하는 서버 선정부; a) 상기 설정된 가동 모드가 액티브/액티브 가동 모드인 경우, 상기 불안정한 상태에 있는 주 서버의 모든 프로세스를 상기 주 서버 선정부에서 선정된 여분 용량을 가진 주 서버에 이중화시키고, b) 상기 설정된 가동 모드가 액티브/스탠바이 가동 모드인 경우, 상기 불안정한 상태에 있는 주 서버의 모든 프로세스를 상기 주 서버 선정부에서 선정된 여분 서버에 이중화시키는 이중화부를 포함한다.

<53> 본 발명에 따른 클러스터 컴퓨터 시스템의 결함 감시 장치의 일측면에 따르면, 상기 다수의 주 서버 각각에 대해 고장으로 인한 서비스의 중단을 감지하는 감지 수단; 상기 감지결과, 임의의 주 서버에 결함이 발생되어 서비스가 중단되면, 주 서버를 임의의 부 서버로 절체한 후, 결함이 발생된 주 서버의 결함 복구 명령 신호를 발생하는 결함 복구 명령 신호 발생 수단; a) 상기 발생된 복구 명령 신호에 따라 주 서버에 대한 모든 기능을 상기 부 서버로 전이하고, b) 부 서버로 전이가 완료되면, 부 서버를 주 서버로 등록하고, 결함이 발생된 상기 주 서버를 가용 서버 리스트로부터 삭제하며, c) 상기 가용 서버 리스트로부터 삭제된 서버의 결함을 복구한 후, 결함이 복구된 서버를 부 서버로 가용 서버 리스트에 등록하는 결함 복구 수단을 포함한다.

<54> 한편, 다수의 주 서버와 여분 서버로 이루어진 클러스터 컴퓨터 시스템의 소프트웨어 가용도 개선 방법을 수행하기 위하여 디지털 처리장치에 의해 실행될 수 있는 명령어들의 프로그램이 유형적으로 구현되어 있으며, 디지털 처리장치에 의해 판독될 수 있는 기록 매체에 따르면, 상기 다수의 주 서버에 대한 시스템 상태 정보를 수집하여 상기 서버들의 이상 상태 유무를 감시하는 단계; 상기 감시결과, 임의의 서버가 불안정한 상태로 판단되면, 여분 서버 또는 여분 용량이 존재하는 다른 주 서버가 존재하는지를 판단하는 단계; 여분 서버 또는 여분 용량을 가진 임의의 주 서버가 존재하는 경우, 현재 설정되어 있는 가동 모드에 따라 불안정한 상태의 주 서버를 여분 서버 또는 여분 용량이 있는 다른 주 서버로

모든 프로세스를 이중화하는 단계; 이중화가 완료되면, 불안정한 상태로 판단된 해당 서버에 재할 동작을 수행하기 위한 시스템 재할 제어 신호를 제공하는 단계를 포함한다.

<55> 또한, 다수의 주 서버와 부 서버로 이루어진 클러스터 컴퓨터 시스템의 결합 감시 방법을 수행하기 위하여 디지털 처리 장치에 의해 실행될 수 있는 명령어들의 프로그램이 유형적으로 구현되어 있으며, 디지털 처리 장치에 의해 판독될 수 있는 기록 매체에 따르면, 상기 다수의 주 서버 각각에 대해 고장으로 인한 서비스의 중단을 감지하는 단계; 상기 감지결과, 임의의 주 서버에 결합이 발생되어 서비스가 중단되면, 주 서버를 임의의 부 서버로 절체한 후, 결합이 발생된 주 서버의 결합 복구 명령을 발생하는 단계; a) 상기 발생된 복구 명령에 따라 주 서버에 대한 모든 기능을 상기 부 서버로 전이하고, b) 여분 서버로의 전이가 완료되면, 여분 서버를 주 서버로 등록하고, 결합이 발생된 상기 주 서버를 가용 서버 리스트로부터 삭제하는 단계; 상기 가용 서버 리스트로부터 삭제된 서버의 결합을 복구한 후, 결합이 복구된 서버를 부 서버로 가용 서버 리스트에 등록하는 단계를 수행한다.

<56> 이하, 본 발명에 따른 클러스터 컴퓨터 시스템의 소프트웨어 가용도 개선 방법 및 그 장치의 바람직한 실시예를 설명하기에 앞서 소프트웨어 재할을 고려한 클러스터 컴퓨터 시스템의 상태 전이 모델에 대하여 간단하게 살펴보기로 하자.

- <57> 도 3은 본 발명에 따른 소프트웨어 재활을 고려한 클러스터 컴퓨터 시스템의 상태 전이 모델을 나타낸 도면이다.
- <58> 도 3에 도시된 바와 같이, 정상 상태에서 가동되고 있는 서버는 $n, n-1, \dots, 1, 0$ 등의 가동중인 서버의 수를 상태 변수(State Parameter)로 가지고 있으며, 장시간 가동으로 인해 성능이 저하된 불안정 상태의 서버는 $u_n, u_{n-1}, \dots, u_2, u_1$ 으로 나타냈다.
- <59> 불안정 상태에서는 λ_r 의 재활률로 재활 작업에 들어가거나, 혹은 $i * \lambda$ 의 고장률(i 는 정상 가동중인 서버의 수)로 고장이 발생하게 된다.
- <60> 또한, 정상 상태에서 불안정 상태로의 변화율은 λ_f 로 표시되며, 이는 소프트웨어의 장기간 가동으로 인한 시스템의 불안정성을 반영한다. 도 2에서 재활 작업 영역(200)의 $r_n, r_{n-1}, \dots, r_1, r_1$ 은 재활 상태를 표시하며, 시스템의 가동을 고의로 중지시켜 재 부팅하는 상황을 나타낸다.
- <61> 클러스터 컴퓨터 시스템의 운영 상태 모델의 수학적 해를 구하기 위해 사용된 가정은 다음과 같다. n 대의 서버로 구성된 클러스터 컴퓨터 시스템에서 각 서버의 고장률(λ)은 동일하며, 고장난 서버를 수리하는 수리율(μ) 또한 모두 같다.
- <62> 클러스터 컴퓨터 시스템에서 소프트웨어 재활에 들어갈 경우, 서버의 가동을 강제적으로 멈추는 재활률(λ_r)은 모든 가동 상태에서 동일하며, 재활 작업률(μ_r)은 서버 수에 무관하다. 클러스터 시스템에서 결함 발생시에 다른 서버로의 작업 전이 시간(Switchover Time)은 극히 짧으므로 무시할 수 있으며, 서

버가 한 대만 가동되는 경우(Simplex System)를 제외하고는 현재 제공되는 서비스의 중단 없이 재활 작업을 수행한다. 마지막으로 도 3의 모든 상태에서 머무는 시간의 길이는 지수 분포를 따른다.

<63> 도 3의 클러스터 컴퓨터 시스템 상태 전이 모델은 상기와 같은 가정하에서 Irreducible Recurrent Nonnull Markov Chain을 형성하므로, 평형 상태에서 확률값을 비교적 쉽게 구할 수 있게 되며, 각 상태에 머물 확률은 아래의 수학적식 4, 5, 6, 7과 같다.

<64>

【수학적식 4】
$$P_{u_{n-i}} = \frac{\lambda_f}{\lambda_r + (n-i)\lambda} P_{n-i}, i=0,1,\dots,n-1$$

<65>

【수학적식 5】
$$P_{r_{n-i}} = \frac{\lambda_r}{\mu} \frac{\lambda_f}{\lambda_r + (n-i)\lambda} P_{n-i}, i=0,1,\dots,n-1$$

<66>

【수학적식 6】
$$P_{n-i} = \left(\frac{\lambda_f}{\mu}\right)^i \prod_{k=0}^{i-1} \left(1 - \frac{\lambda_f}{\lambda_r + (n-k)\lambda}\right) * P_n, i=1,2,\dots,n$$

<67>

【수학적식 7】
$$P_n = \left[1 + \sum_{i=1}^n \left(\left(\frac{\lambda_f}{\mu}\right)^i \prod_{k=0}^{i-1} \left(1 - \frac{\lambda_f}{\lambda_r + (n-k)\lambda}\right) \right) \left(1 + \frac{\lambda_f}{\mu}\right) + \left(\sum_{i=1}^n \frac{\lambda_f}{\lambda_r + (n-i)\lambda} \left(\frac{\lambda_f}{\mu}\right)^i \prod_{k=0}^{i-1} \left(1 - \frac{\lambda_f}{\lambda_r + (n-k)\lambda}\right) + \frac{\lambda_f}{\lambda_r + n\lambda} \right) \right]^{-1}$$

<68> 도 4는 본 발명에 따른 이중화된 클러스터 시스템의 소프트웨어 재활 기법 적용 예시도이다.

<69> 도 4에 도시된 바와 같이 불안정 상태에서 가동되고 있는 2대의 서버(u_2)는 2λ 의 고장률로 하드웨어적인 고장이 발생하며, 이 경우 λ 는 서버의 MTTF(Mean Time To Failure)로부터 계산할 수 있다. 두 대의 서버가 모두 가동을 멈추는 고장 상태에서는 μ 의 변화율로 고장이 수리되며, 이 값은 고장 수리 능력의 척도인 MTTR(Mean Time To Repair)에서 구할 수 있다. 서버의 장시간 가동으로 인한 소프트웨어 노화로 인해 서버의 성능이 저하되는 불안정 상태에 있을 경우 고의로 시스템의 가동을 멈추는 재할 상태(300) r_2 , r_1 로 가거나 혹은 고장 상태로 진행하게 된다.

<70> 결국, 도 2에 도시된 종래 기술에서는 장시간 가동된 소프트웨어의 불안정성을 고려하지 않은 상태의 전이 모델을 표시한 것으로, 이 경우 불안정 상태와 소프트웨어 재할 상태에 대한 표현이 없는 것이다. 즉, 도 3의 클러스터 컴퓨터 시스템 상태 전이 모델에서 도출한 확률로부터 가용도, 다운 타임 및 손실 비용을 정의한 경우가 상기한 수학적 1, 2, 3이 되는 것이다.

<71> 이하, 본 발명에 따른 클러스터 컴퓨터 시스템의 소프트웨어 가용도 개선 방법 및 그 장치에 대한 바람직한 실시예에 대하여 첨부한 도면을 참조하여 상세하게 설명해 보기로 하자.

<72> 도 5는 본 발명에 따른 소프트웨어 가용도 개선 장치를 포함하는 클러스터 컴퓨터 시스템의 구성을 나타낸 도면으로서, 소프트웨어 재할 기법의 적용 대상이 되는 고 가용성 클러스터 컴퓨터 시스템의 구조를 표시하며 그 구성은 클러스터링 모듈(501), 소프트웨어 재할 모듈(502), 결합 허용 모듈(503)로 구성된다.

- <73> 클러스터링 모듈(501)에서는 여러대의 컴퓨터를 연결해 고 가용성 클러스터 시스템을 구축하는 기능을 제공하며, 이론상 연결될 수 있는 서버 수는 제한이 없다. 클러스터 컴퓨터 시스템의 가동 모드에 따라 액티브/스탠바이(Active/Standby), 액티브/액티브(Active/Active)방식으로 구분되며, 전자의 경우 실제로 서비스에는 참가하지 않는 여분 서버(505)가 있는 경우이며, 후자는 클러스터를 구성하는 모든 서버가 서비스에 참여하면서 동시에 상호 간에 여분 서버(505)역할을 수행하는 모드이다.
- <74> 그리고, 클러스터링 모듈(501)은 클러스터 컴퓨터 시스템을 구성하고 있는 개개 서버의 작업 부하를 조절하는 부하 분배 기능을 수행하고, 소프트웨어 재활 모듈(502)과 재활 작업에 필요한 자료를 송,수신한다.
- <75> 소프트웨어 재활 모듈(502)에서는 클러스터 컴퓨터 시스템의 서버를 대상으로 시스템 운영 파라미터에 따른 분석 결과에 근거하여 서버의 소프트웨어적인 불안정성을 파악한 후, 강제적으로 불안정 상태에 있는 서버의 동작을 중지시키는 명령을 생성한다. 이 재활 명령은 임의의 불안정한 상태에 있는 서버의 상태를 결합 허용 모듈(503)과 클러스터링 모듈(501)의 협조를 받아 결합 발생 가능성이 낮은 가동 초기 상태로 복귀시킨다. 이때 재활을 수행하는 기준, 방법 및 절차는 클러스터 컴퓨터 시스템의 응용 분야별로 적절하게 설정될 수 있다.
- <76> 그리고, 결합 허용 모듈(503)에서는 클러스터 컴퓨터 시스템 서버들의 결합을 감지하고, 결합 발생 서버의 절체 및 결합 복구 기능을 담당한다. 결합 감지 기능을 수행하기 위해서 다양한 결합 감지 기법 예를 들어, Heart Beat, Watch Dog 등이 사용될 수 있으며, 결합이 발생한 서버를 절체할 때에는 체크 포인팅

등의 결합 허용 기법이 활용된 주 서버(504)의 작업 상태를 대기중인 여분 서버(505) 혹은 여유가 있는 다른 서버로 전달하는 것이다.

<77> 또한, 도 5에서는 모두 $n+k$ 개의 서버로 구성된 클러스터 컴퓨터 시스템의 예를 나타냈으며, n 대의 주 서버(504), k 대의 여분 서버(505)로 구성된 모습을 나타내고 있다. 일반적으로 재할 대상이 되는 서버에서 수행되는 모든 프로세스는 수행이 중지되며, 재할 작업을 마친 서버는 결합 발생 가능성이 낮은 상태를 가지고 재 출발하게 된다. 재할 명령이 수행되기 전에 클러스터링 모듈(501)은 재할 대상이 되는 서버로는 작업 부하를 배분하지 않으며, 재할 작업이 수행된 결합 발생 확률이 작은 건강한 상태의 서버 정보가 통보되어 작업 부하를 재 배정 받도록 구성된다. 따라서, 재할 대상 서버에서 수행되는 프로세스를 대상으로 하는 재할이 아니라 개개 서버 수준에서의 재할을 수행하게 되며, 이 경우 프로세스 수준의 재할을 수행할 경우 발생하는 데이터의 복잡성, 데이터 구조 설계 등의 오버헤드 소요 비용을 크게 줄일 수 있게 된다.

<78> 도 5와 같은 (n,k) 클러스터 컴퓨터 시스템의 경우 대기중인 특정 서버에 재할 명령의 대상이 되는 서버의 모든 프로세스를 작업전이(Switchover)한 후, 재할 작업을 수행할 경우 가용도 저하로 인한 손실 비용이 발생하지 않게 된다.

<79> 고 가용도 클러스터 시스템을 여분 서버 없이 구성할 경우 성능 대비 비용 효과가 상승하지만 여분 서버를 둘 경우에는 성능은 저하되지만 서비스에 대한 가용도가 증가하는 트레이드 오프(Trade-off)가 발생한다.

- <80> 도 5에 도시된 고 가용성 클러스터 컴퓨터 시스템에서 클러스터링 모듈에 대한 상세 구성 및 동작에 대하여 도 6을 참조하여 살펴보자.
- <81> 도 5에 도시된 클러스터링 모듈의 상세 구성을 나타낸 도면으로서, 클러스터링 모듈(501)은 이중화 구조의 부하 분배기(601)와 클러스터 제어기(602)로 구성될 수 있다.
- <82> 클러스터링 모듈(501)내의 이중화 구조의 부하 분배기(601)는 각 클러스터 서버에 부하를 균등히 배분하는 역할을 수행함과 동시에 소프트웨어 재활 모듈(502)의 명령을 직접 수행한다.
- <83> 특정 서버의 연속 가동 시간과 현재의 작업 부하 등을 고려한 후, 재활 작업의 대상이 되는 서버를 선정하며, 선정된 서버는 부하 분배기(601)의 가용 서버 리스트에서 제외한다, 이후 응용 분야별로 최적 재활 조건이 성립할 때 재활 작업 명령을 내리게 되는 것이다.
- <84> 그리고, 도 5에 도시된 소프트웨어 재활 모듈(502)의 상세 구성은 도 7에 도시되어 있으며, 도 7을 참조하여 그 구성을 살펴보면, 재활 명령 생성기(701), 시스템 상태 수집기(702) 및 시스템 감시기(703)로 구성될 수 있다.
- <85> 재활 명령 생성기(701)에서는 소프트웨어 재활 명령을 클러스터 컴퓨터 시스템의 가동 상태(작업 부하, 연속 가동 시간 등)를 고려한 후, 동작으로 내릴 수 있다. 한편, 클러스터 컴퓨터 시스템의 가동 상태와는 무관한 주기가 이미 확정된 정적 소프트웨어 재활도 가능하며, 이 때에는 주기적으로 소프트웨어 재활 작업이 수행되는 것이다. 재활 작업은 백 그라운드 데몬 프로세스로 수행되며 정

적인 소프트웨어 재할 작업 수행시에는 UNIX 환경의 경우 cron과 같은 명령을 이용해 미래의 주기적인 재할 작업 시간 및 조건을 예약할 수 있는 것이다.

<86> 시스템 상태 수집기(702)는 클러스터 서버의 현재 상태에 관한 정보 예를 들면, 서버의 불안정 상태 정보, 고장 상태 정보, 작업 전이 상태 정보 등을 관리한다. 이러한 상태 정보와 시스템 감시기(703)에서 파악된 클러스터 서버내의 프로세스에 관련된 정보 예를 들면, 작업 부하, 연속 가동 시간, 메모리 사용량 정보 등은 재할 명령 생성기(701)에 입력되어 재할 정책 수립에 사용된다.

<87> 한편, 도 5에 도시된 결합 허용 모듈(503)에 대하여 도 8을 참조하여 상세하게 살펴보자. 도 8은 도 5에 도시된 결합 허용 모듈의 상세 구성을 나타낸 도면으로서, 결합 허용 모듈(503)은 결합 감지기(801), 결합 복구기(802) 및 결합 절체기(803)로 구성될 수 있다.

<88> 상기 결합 감지기(801)에서는 서버의 고장으로 인한 서비스 중단을 감지한다.

<89> 서버의 결합이 감지되면, 감지된 신호는 결합 절체기(803)로 제공되는데, 결합 절체기(803)는 상기 결합 감지기(801)에서 결합이 감지된 서버를 클러스터 컴퓨터 시스템으로부터 분리/절체하게 되는 것이다.

<90> 상기 결합 절체기(803)에서 결합이 감지된 서버를 클러스터 컴퓨터 시스템으로부터 절체가 이루어지면, 결합 복구기(802)에서는 여분 서버로 주 서버의 기능을 전이하는 것이다. 고의적으로 서버의 가동을 중단하는 경우 재할 명령을 받은 서버는 결합 허용 모듈(503)의 이중화 명령을 수용하여 재할 대상 서버의 모

든 프로세스 관련 정보를 여분 서버에게 전달하여 주 서버의 프로세스가 완전 이
중화 되도록 하는 것이다.

<91> 상기와 같은 구성을 갖는 본 발명에 따른 클러스터 컴퓨터 시스템의 소프트
웨어 가용도 개선 장치에 대한 구체적인 동작을 첨부한 도 9를 참조하여 설명해
보자.

<92> 도 9는 본 발명에 따른 클러스터 컴퓨터 시스템의 소프트웨어 가용도 개선
장치의 연결 구성을 나타낸 도면으로서, 도 6,7,8과 그 내부 구성은 서로 동일하
기 때문에 구성 설명은 생략하기로 하고, 서버가 불안정한 상태인 경우 복구하는
동작 과 서버의 결함 발생시 복구하는 동작을 각각 구분하여 설명하기로 한다.

<93> 먼저, 서버가 불안정한 상태 인 경우에 대하여 살펴보면, 소프트웨어 재활
모듈(502)의 시스템 감시기(703)에서는 주 서버(504)들의 작업 부하, 연속 가동
시간, 메모리 사용량, 버퍼 사용량 등을 감시하여 감시 결과 정보를 시스템 상태
수집기(702)로 제공한다.

<94> 시스템 상태 수집기(702)는 시스템 감시기(703)에서 제공되는 각 서버들의
감시 정보들을 이용하여 주 서버(504)의 소프트웨어 불안정 상태, 고장 상태, 작
업 전이 상태 등을 파악하여 재활 명령 생성기(701)로 제공한다.

<95> 재활 명령 생성기(701)는 시스템 상태 수집기(702)에서 제공되는 주 서버
(504)의 상태 정보에 따라 불안정한 주 서버(504)가 존재하는지를 판단한 후, 불
안정한 상태의 주 서버(504)가 존재하는 경우 해당 주 서버(504)의 불안정한 소
프트웨어의 복구를 위한 즉, 불안정한 상태에 있는 주 서버(504)의 재활을 위한

재활 명령을 생성하여 클러스터링 모듈(501)내 부하 분배기(601)로 통보한다.

즉, 부하 분배기(601)에 재활 대상이 되는 불안정한 상태의 임의의 주 서버를 통보하게 된다.

<96> 부하 분배기(601)는 해당 서버의 재활을 위해 클러스터 제어기(602)에 재활 제어 신호를 제공하게 된다.

<97> 따라서, 클러스터 제어기(602)는 여분 서버(505) 혹은 여분 용량의 주 서버(504)가 존재하는지를 판단한 후, 여분 서버 혹은 여분 용량이 있는 주 서버(504)가 존재하는 경우, 현재 설정된 모드를 판단하고, 현재 설정된 모드에 따라 불안정한 상태에 있는 주 서버의 재활을 위한 재활 제어 신호를 결합 허용 모듈(503)의 결합 복구기(802)로 제공하게 된다.

<98> 결합 허용 모듈(503)내 결합 복구기(802)는 클러스터 제어기(602)로부터 제공되는 제어 신호에 따라 상기 불안정한 상태에 있는 주 서버의 프로세스를 여분 서버 혹은 여분 용량이 존재하는 주 서버로 이중화시키게 되는 것이다. 여기서, 상기 설정 모드는 관리자가 설정하는 모드로서, 현재 설정된 모드가 액티브/스탠바이(Active/Standby)모드인 경우 결합 복구기(802)는 임의의 여분 서버를 선정하여 불안정한 상태에 있는 주 서버의 모든 프로세스를 상기 선정된 임의의 여분 서버로 이중화시킨다.

<99> 한편, 현재 설정된 모드가 액티브/액티브(Active/Active)모드로 설정되어 있는 경우, 결합 복구기(802)는 여분 용량을 가진 임의의 서버에 불안정한 상태에 놓여 있는 주 서버의 모든 프로세스를 이중화시키게 되는 것이다. 이와 같이 이중화 작업이 완료된 상태에서도 소프트웨어 재활 모듈(502)의 시스템 감시기

(703)에서는 불안정한 상태의 주 서버 즉, 재할 대상이 되는 주 서버의 작업 부하, 연속 가동 시간, 메모리 사용량, 버퍼 사용량 등을 감시하게 된다. 따라서, 클러스터링 모듈(501)의 부하 분배기(601)에서는 소프트웨어 재할 모듈(502)로부터 제공되는 재할 대상이 되는 주 서버의 작업 부하, 연속 가동 시간 등의 정보를 고려하여 재할 명령 수행 여부를 계속적으로 판단하게 된다.

<100> 계속적으로 재할 대상이 되는 주 서버의 시스템 안정 상태가 불안정한 상태를 유지하게 되면, 클러스터 제어기(602)는 부하 분배기(601)의 가용 서버 리스트에서 재할 대상이 되는 주 서버를 제외시키고, 재할 대상이 되는 주 서버와, 여분 서버 또는 여분 용량이 있는 서버를 주 서버로 전환시키게 되는 것이다.

<101> 그리고, 클러스터 제어기(602)는 재할 대상이 되는 주 서버에 재할 명령을 전송하게 되면, 해당 주 서버는 소프트웨어의 재할을 수행하게 되는 것이다. 여기서, 소프트웨어 재할은 파일 시스템 청소, 버퍼 청소, 메모리 청소, 재 부팅 등을 통해 소프트웨어를 재할하게 되는 것이다.

<102> 이와 같이 재할이 완료된 주 서버는 클러스터 제어기(602)로 재할 완료 정보를 제공하게 되고, 클러스터 제어기(602)는 이 정보를 수신하여 부하 분배기의 가용 서버 리스트에 등록하여 추후 여분 서버로 이용하게 되는 것이다.

<103> 이어, 주 서버(504)의 결함이 발생되어 서비스가 중단된 경우 결함이 발생한 주 서버의 결함을 복구하는 동작에 대하여 설명해 보자.

<104> 먼저, 주 서버의 결함 감지 및 복구 동작은 상기한 서버가 불안정한 상태인 경우 해당 서버의 소프트웨어 재할 동작과는 무관하게 동시에 진행된다.

- <105> 도 9에 도시된 결합 허용 모듈(503)내의 결합 감지기(801)는 다수의 주 서버(504)들의 결합 여부를 감지하게 된다.
- <106> 감지 결과, 다수의 주 서버(504) 중 임의의 주 서버에 결합이 발생되었음을 감지하게 되면, 감지 신호를 결합 절체기(803)로 제공한다.
- <107> 결합 절체기(803)는 결합 감지기(801)에서 결합이 감지된 주 서버를 여분 서버로 절체하게 되고, 그 결과 신호 및 결합이 발생된 주 서버의 복구 명령 신호를 결합 복구기(802)로 제공한다. 여기서, 절체된 여분 서버가 주 서버의 역할을 수행하게 되는 것이다.
- <108> 따라서, 결합 복구기(802)는 결합이 발생된 주 서버의 결합을 복구하게 되는 것이다.
- <109> 결합 복구가 완료되면, 결합이 제거된 해당 서버를 클러스터 제어기(602)를 통해 부하 분배기(601)의 가용 서버 리스트에 등록을 하는 것이다.
- <110> 이상에서 설명한 본 발명에 따른 클러스터 컴퓨터 시스템의 소프트웨어 가용도 개선 장치의 동작과 상응하는 본 발명에 따른 클러스터 컴퓨터 시스템의 소프트웨어 가용도 개선 방법에 대하여 서버가 불안정한 상태인 경우 복구하는 방법과, 서버의 결합 발생 시(하드웨어의 결합으로 서비스 중단시) 복구하는 방법을 각각 구분하여 도 10과 도 11를 참조하여 살펴보기로 한다.
- <111> 도 10은 본 발명에 따른 클러스터 컴퓨터 시스템의 소프트웨어 가용도 개선 방법에 있어서, 서버의 불안정한 상태(소프트웨어의 불안정 상태) 발생시 이를 복구하는 방법에 대한 동작 플로우차트를 나타낸 도면이다.

- <112> 먼저, 주 서버들의 작업 부하, 연속 가동 시간, 메모리 사용량, 버퍼 사용량 등을 감시하고, 감시된 각 서버들의 감시 정보들을 이용하여 주 서버의 소프트웨어 불안정 상태, 고장 상태, 작업 전이 상태 등을 파악하게 되는 것이다.
- <113> 이렇게 파악된 주 서버의 상태 정보에 따라 불안정한 주 서버가 존재하는지를 판단한 후, 불안정한 상태의 주 서버가 존재하는 경우 해당 주 서버의 불안정한 소프트웨어의 복구를 위한 즉, 불안정한 상태에 있는 주 서버의 재활을 위한 재할 명령을 생성하여 클러스터링 모듈 내 부하 분배기로 통보한다(S101). 즉, 부하 분배기(601)에 재할 대상이 되는 불안정한 상태의 임의의 주 서버를 통보하게 된다.
- <114> 이어, 상기 불안정한 상태에 있는 주 서버의 재활을 위한 여분 서버 혹은 여분 용량이 있는 주 서버가 존재하는지를 판단한다(S102).
- <115> 판단 결과, 여분 서버 혹은 여분 용량이 있는 주 서버가 존재하는 경우, 현재 설정된 모드를 판단하고, 현재 설정된 모드에 따라 불안정한 상태에 있는 주 서버의 모든 프로세스를 상기 여분 서버 또는 여분 용량을 가진 주 서버에 이중화시키게 된다.
- <116> 여기서, 상기 설정 모드는 관리자가 설정하는 모드로서, 현재 설정된 모드가 액티브/스탠바이(Active/Standby)모드인 경우, 임의의 여분 서버를 선정하여 불안정한 상태에 있는 주 서버의 모든 프로세스를 상기 선정된 임의의 여분 서버로 이중화시킨다.

- <117> 한편, 현재 설정된 모드가 액티브/액티브(Active/Active)모드로 설정되어 있는 경우에는 여분 용량을 가진 임의의 서버에 불안정한 상태에 놓여 있는 주 서버의 모든 프로세스를 이중화시키게 되는 것이다(S103).
- <118> 이와 같이 이중화 작업이 완료된 상태에서도 불안정한 상태의 주 서버 즉, 재활 대상이 되는 주 서버의 작업 부하, 연속 가동 시간, 메모리 사용량, 버퍼 사용량등을 감시하게 되고, 감시 정보 즉, 재활 대상이 되는 주 서버의 작업 부하, 연속 가동 시간 등의 정보를 고려하여 재활 명령 수행 여부를 계속적으로 판단하게 된다(S104).
- <119> 계속적으로 재활 대상이 되는 주 서버의 시스템 안정 상태가 불안정한 상태를 유지하게 되면, 클러스터링 모듈내 부하 분배기의 가용 서버 리스트에서 재활 대상이 되는 주 서버를 제외시키고, 여분 서버 또는 여분 용량이 있는 서버를 주 서버로 전환시키게 되는 것이다(S105).
- <120> 이어, 재활 대상이 되는 주 서버에 재활 명령을 전송하여, 주 서버로 하여금 소프트웨어의 재활을 수행하게 하는 것이다. 여기서, 소프트웨어 재활은 파일 시스템 청소, 버퍼 청소, 메모리 청소, 재 부팅 등을 통해 소프트웨어를 재활하게 되는 것이다.
- <121> 이와 같이 재활이 완료된 주 서버는 클러스터 제어기를 통해 부하 분배기에 가용 서버 리스트 등록 정보를 제공하고 따라서, 부하 분배기는 해당 서버를 가용 서버 리스트에 등록을 하게 되는 것이다(S106).

- <122> 도 11은 본 발명에 따른 클러스터 컴퓨터 시스템의 소프트웨어 가용도 개선 방법에 있어서, 서버의 결함(하드웨어 결함으로 서비스 중단시)시 이를 복구하는 방법에 대한 동작 플로우차트를 나타낸 도면이다.
- <123> 먼저, 결함 감지기를 통해 다수의 주 서버들의 결함 여부를 감지하고, 임의의 주 서버에 결함이 발생되었는지를 판단한다(S201).
- <124> 판단 결과, 다수의 주 서버 중 임의의 주 서버에 결함이 발생되었음을 감지하게 되면, 결함이 감지된 주 서버를 여분 서버로 절체하여 절체된 여분 서버가 주 서버의 역할을 수행하도록 한다(S202).
- <125> 이어, 여분 서버가 주 서버의 동작을 수행하는 도중 결함이 발생한 주 서버는 결함을 복구하는 동작을 수행한다. 이어, 결함이 발생한 주 서버의 결함이 모두 복구되었는지를 판단한다(S203).
- <126> 해당 서버의 결함 복구가 완료되면, 결함이 제거된 해당 서버를 클러스터링 모듈내 부하 분배기의 가용 서버 리스트에 등록을 함으로써, 결함 허용 동작이 완료되는 것이다(S204).

【발명의 효과】

- <127> 상기한 바와 같은 본 발명에 따른 클러스터 컴퓨터 시스템의 소프트웨어 가용도 개선 방법 및 그 장치는, 시스템의 결함 발생 이후에 수동적으로 대처하는 기존의 결함 허용 방법에 비해 결함이 발생하기 전에 이를 미연에 방지하는 능동적/예방적 차원의 결함 허용이 가능한 것이다.

<128> 이와 같은 본 발명은, 앞으로 다가올 인터넷 기반 비즈니스 시대에 반드시 필요한 기반 기술의 하나이며, 인터넷 환경하에서의 신뢰도 높은 데이터 서비스를 제공하기 위한 기본 요소이다. 소프트웨어 재활 방법은, 관련 시스템에 탑재된 소프트웨어 결함을 미연에 방지하여 최근 늘어나는 시스템 유지 보수 비용을 절감하여 제품 경쟁력을 높일 수 있다.

<129> 또한, 대용량 트랜잭션 서비스 관련 기반 기술 분야는 모든 고급 컴퓨터 설계 기술의 총아라 할 수 있으므로 본 발명에 따른 재활 기법은 여타 많은 컴퓨터 시스템 설계 관련 분야의 가용도 개선을 위한 기반 기술 축적에 초석이 될 것이다.

<130> 특히 멀티미디어 이동 컴퓨팅에서 사용되는 소프트웨어는 통신, 단절, 데이터 유실 등으로 인한 노화 진행이 일반 소프트웨어 보다 상당히 빠르게 진행되기 때문에 소프트웨어 재활에 의한 결함 예방 방법은 대규모 멀티미디어 이동 컴퓨팅 시스템에 사용될 가능성이 높다고 볼 수 있다.

【특허청구범위】**【청구항 1】**

다수의 주 서버와 여분 서버로 이루어진 클러스터 컴퓨터 시스템의 소프트웨어 가용도 개선 방법에 있어서,

상기 다수의 주 서버에 대한 시스템 상태 정보를 수집하여 상기 서버들의 이상 상태 유무를 감시하는 단계;

상기 감시결과, 임의의 서버가 불안정한 상태로 판단되면, 여분 서버 또는 여분 용량이 존재하는 다른 주 서버가 존재하는지를 판단하는 단계;

여분 서버 또는 여분 용량을 가진 임의의 주 서버가 존재하는 경우, 현재 설정되어 있는 가동 모드에 따라 불안정한 상태의 주 서버를 여분 서버 또는 여분 용량이 있는 다른 주 서버로 모든 프로세스를 이중화하는 단계;

이중화가 완료되면, 불안정한 상태로 판단된 해당 서버에 재할 동작을 수행하기 위한 시스템 재할 제어 신호를 제공하는 단계를 포함하는 클러스터 컴퓨터 시스템의 소프트웨어 가용도 개선 방법.

【청구항 2】

제1항에 있어서,

상기 시스템 상태 정보는, 주 서버의 작업 부하, 연속 가동 시간, 메모리 사용량, 버퍼 사용량 정보 중 적어도 하나의 정보를 포함하는 클러스터 컴퓨터 시스템의 소프트웨어 가용도 개선 방법.

【청구항 3】

제1항에 있어서,
상기 모든 프로세스를 이중화하는 단계에서의 설정 모드는,
실제로 서비스에 참가하지 않는 여분 서버가 존재하여 이중화시 이용하는
액티브/스탠바이 모드;

클러스터를 구성하는 모든 서버가 서비스에 참여하면서 동시에 상호간에 여
분 서버 역할을 수행하는 액티브/액티브 모드를 포함하는 클러스터 컴퓨터 시스
템의 소프트웨어 가용도 개선 방법.

【청구항 4】

제1항에 있어서,
상기 모든 프로세스를 이중화하는 단계는,
상기 현재 설정된 모드가 액티브/스탠바이 모드인 경우, 임의의 여분 서버
를 선정하는 단계;

상기 불안정한 상태에 있는 주 서버의 모든 프로세스를 상기 선정된 여분
서버에 이중화하는 단계를 포함하는 클러스터 컴퓨터 시스템의 소프트웨어 가용
도 개선 방법.

【청구항 5】

제1항 또는 제3항에 있어서,

상기 모든 프로세스를 이중화하는 단계는,

상기 현재 설정된 모드가 액티브/액티브 모드인 경우, 여분 용량을 가진 임의의 주 서버를 선정하는 단계;

상기 불안정한 상태에 있는 주 서버의 모든 프로세스를 상기 선정된 여분 용량을 가진 주 서버에 이중화하는 단계를 포함하는 클러스터 컴퓨터 시스템의 소프트웨어 가용도 개선 방법.

【청구항 6】

제1항에 있어서,

상기 서버의 재할 동작을 수행하는 단계는,

재할 대상이 되는 주 서버의 이중화가 완료되면, 재할 대상이 되는 주 서버의 작업 부하, 연속 가동 시간에 따라 재할 명령 수행 여부를 판단하는 단계;

상기 판단 결과 재할 명령 수행이 이루어져야 하는 경우, 가용 서버 리스트에 상기 재할 대상이 되는 주 서버의 리스트를 삭제하는 단계;

상기 이중화된 여분 서버를 주 서버로 전환한 후, 상기 재할 대상이 되는 주 서버의 재할 동작을 수행하는 단계;

재할 동작이 완료되면, 재할 동작이 완료된 주 서버를 가용 서버 리스트에 여분 서버로 등록하는 단계를 포함하는 클러스터 컴퓨터 시스템의 소프트웨어 가용도 개선 방법.

【청구항 7】

제6항에 있어서,

상기 재활 대상이 되는 주 서버의 재활 동작은, 파일 시스템 청소, 버퍼 청소, 메모리 청소, 재 부팅 동작을 포함하는 클러스터 컴퓨터 시스템의 소프트웨어 가용도 개선 방법.

【청구항 8】

다수의 주 서버와 여분 서버로 이루어진 클러스터 컴퓨터 시스템의 소프트웨어 가용도 개선 장치에 있어서,

상기 다수의 주 서버에 대한 시스템 상태 정보를 수집하여 각 서버의 불안정한 상태를 파악하는 시스템 감시 수단;

a) 상기 시스템 감시 수단에서 시스템 감시 결과, 임의의 주 서버가 불안정한 상태인 경우, 현재 설정되어 있는 가동 모드에 따라 불안정한 상태의 주 서버를 여분 서버 또는 여분 용량이 있는 다른 주 서버로 모든 프로세스를 이중화시키기 위한 이중화 제어 신호를 제공하고,

b) 상기 불안정한 상태의 주 서버의 시스템 상태가 일정 시간동안 지속되는 경우 해당 주 서버에 시스템 재활을 위한 재활 신호를 제공하는 클러스터 제어 수단;

상기 클러스터 제어수단에서 제공되는 설정된 모드에 대한 이중화 제어 신호에 따라 불안정한 상태의 주 서버의 모든 프로세스를 여분 서버 또는 여분 용

량이 존재하는 다른 주 서버로 이중화시키는 이중화 수단을 포함하는 클러스터 컴퓨터 시스템의 소프트웨어 가용도 개선 장치.

【청구항 9】

제8항에 있어서,

상기 시스템 감시 수단은,

상기 주 서버들의 시스템 상태를 감시하여 각 서버들의 상태 정보를 수집하는 시스템 상태 정보 수집부;

상기 시스템 상태 정보 수집부에서 수집된 시스템 상태 정보에 따라 불안정한 주 서버가 존재하는지를 판단하고, 불안정한 상태의 주 서버가 존재하는 경우 해당 주 서버의 불안정한 소프트웨어의 재활을 위한 재활 명령 신호를 생성하여 상기 이중화 수단으로 제공하는 재활 명령 생성부를 포함하는 클러스터 컴퓨터 시스템의 소프트웨어 가용도 개선 장치.

【청구항 10】

제8항에 있어서,

상기 시스템의 상태 정보는, 서버의 작업 부하, 연속 가동 시간, 메모리 사용량, 버퍼 사용량 정보 중 적어도 하나의 정보를 포함하는 클러스터 컴퓨터 시스템의 소프트웨어 가용도 개선 장치.

【청구항 11】

제8항에 있어서,

상기 클러스터 제어수단은,

상기 이중화 수단에서 상기 불안정한 상태의 주 서버가 여분 서버 또는 여분 용량을 가진 다른 주 서버로 이중화된 경우 상기 불안정 상태의 주 서버를 가용 서버 리스트로부터 삭제시키고, 상기 재활 신호에 따라 불안정한 상태의 주 서버의 재활이 완료되면 재활이 완료된 주 서버를 가용 서버 리스트로 재 등록하는 등록수단을 포함하는 클러스터 컴퓨터 시스템의 소프트웨어 가용도 개선 장치.

【청구항 12】

제8항에 있어서,

클러스터 제어 수단에 설정되어 있는 가동 모드는,

실제로 서비스에 참가하지 않는 여분 서버가 존재하여 이중화 시 상기 여분 서버를 이용하는 액티브/스탠바이 모드;

클러스터를 구성하는 모든 서버가 서비스에 참여하면서 동시에 상호간에 여분 서버 역할을 수행하는 액티브/액티브 모드를 포함하는 클러스터 컴퓨터 시스템의 소프트웨어 가용도 개선 장치.

【청구항 13】

제8항 또는 12항에 있어서,

상기 이중화 수단은,

상기 클러스터 제어 수단에 설정된 가동 모드에 따라 여분 서버 또는 여분 용량을 가진 임의의 주 서버를 선정하는 서버 선정부;

a) 상기 설정된 가동 모드가 액티브/액티브 가동 모드인 경우, 상기 불안정한 상태에 있는 주 서버의 모든 프로세스를 상기 주 서버 선정부에서 선정된 여분 용량을 가진 주 서버에 이중화시키고,

b) 상기 설정된 가동 모드가 액티브/스탠바이 가동 모드인 경우, 상기 불안정한 상태에 있는 주 서버의 모든 프로세스를 상기 주 서버 선정부에서 선정된 여분 서버에 이중화시키는 이중화부를 포함하는 클러스터 컴퓨터 시스템의 소프트웨어 가용도 개선 장치.

【청구항 14】

다수의 주 서버와 여분 서버로 이루어진 클러스터 컴퓨터 시스템의 소프트웨어 가용도 개선 방법을 수행하기 위하여 디지털 처리장치에 의해 실행될 수 있는 명령어들의 프로그램이 유형적으로 구현되어 있으며, 디지털 처리장치에 의해 판독될 수 있는 기록 매체에 있어서,

상기 다수의 주 서버에 대한 시스템 상태 정보를 수집하여 상기 서버들의 이상 상태 유무를 감시하는 단계;

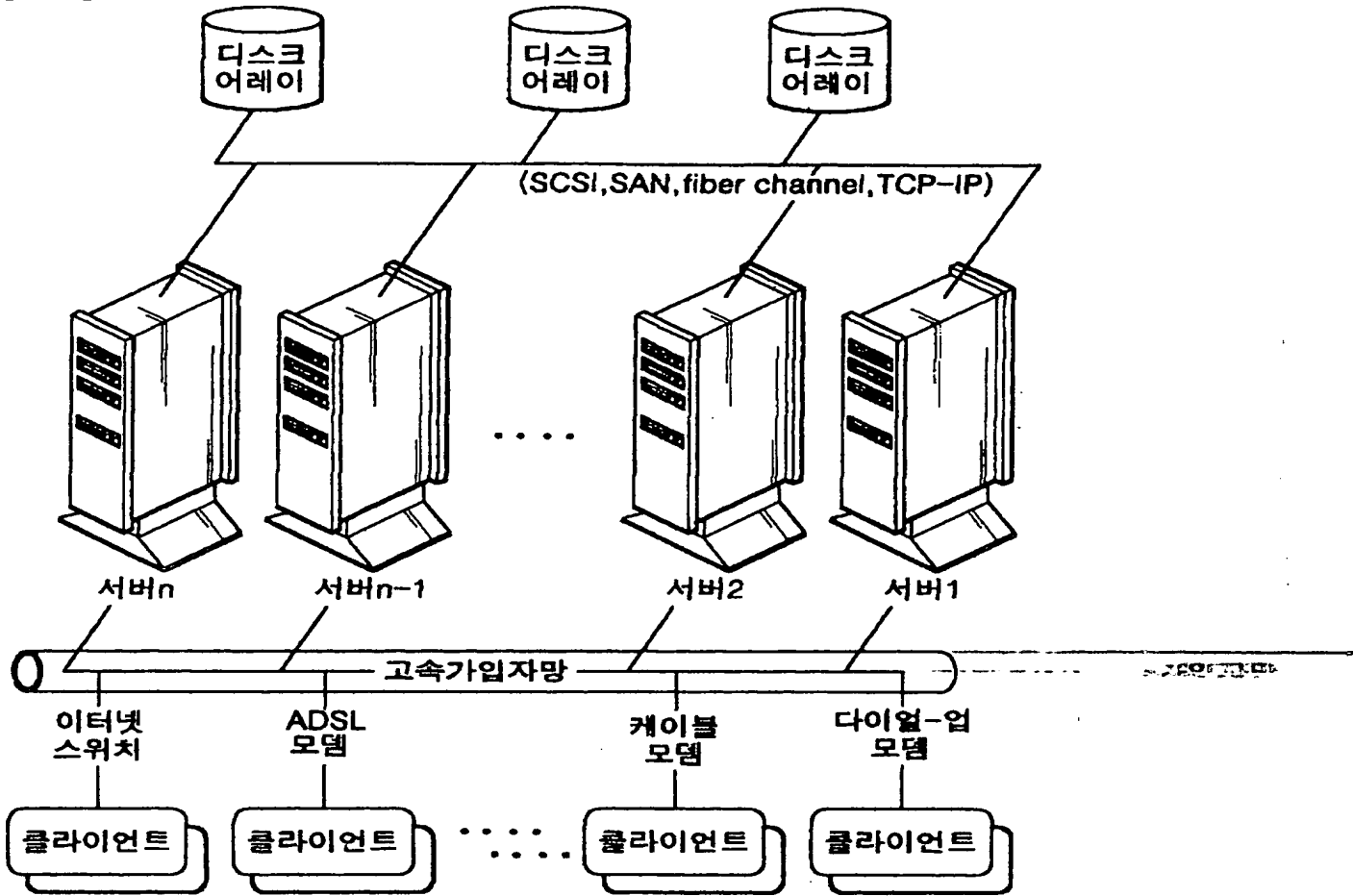
상기 감시결과, 임의의 서버가 불안정한 상태로 판단되면, 여분 서버 또는 여분 용량이 존재하는 다른 서버가 존재하는지를 판단하는 단계;

여분 서버 또는 여분 용량을 가진 임의의 주 서버가 존재하는 경우, 현재 설정되어 있는 가동 모드에 따라 불안정한 상태의 주 서버를 여분 서버 또는 여분 용량이 있는 다른 주 서버로 모든 프로세스를 이중화하는 단계;

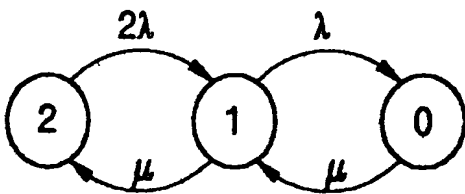
이중화가 완료되면, 불안정한 상태로 판단된 해당 서버에 재할 동작을 수행하기위한 시스템 재할 제어 신호를 제공하는 단계를 수행하는 기록 매체.

【도면】

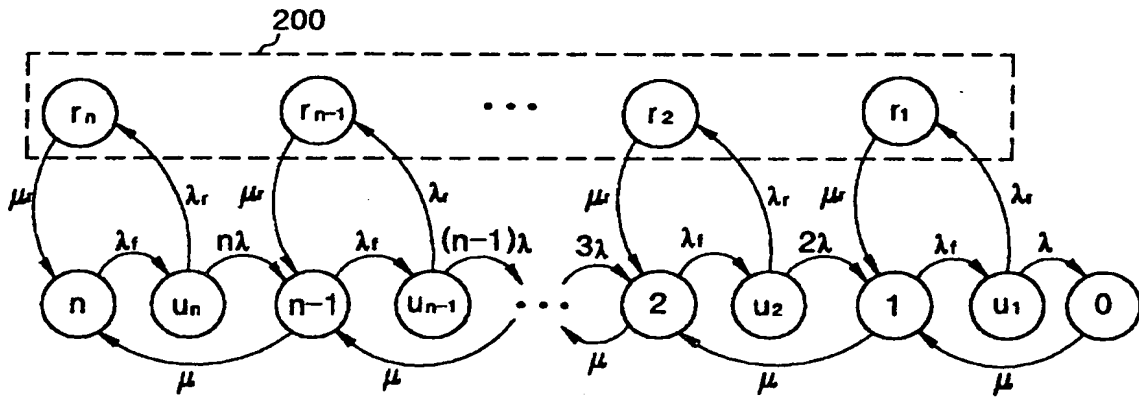
【도 1】



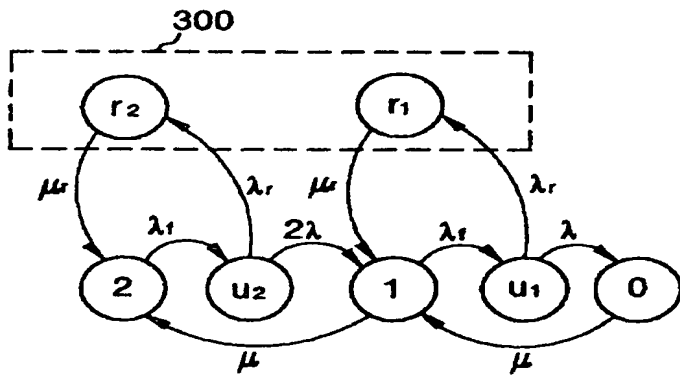
【도 2】



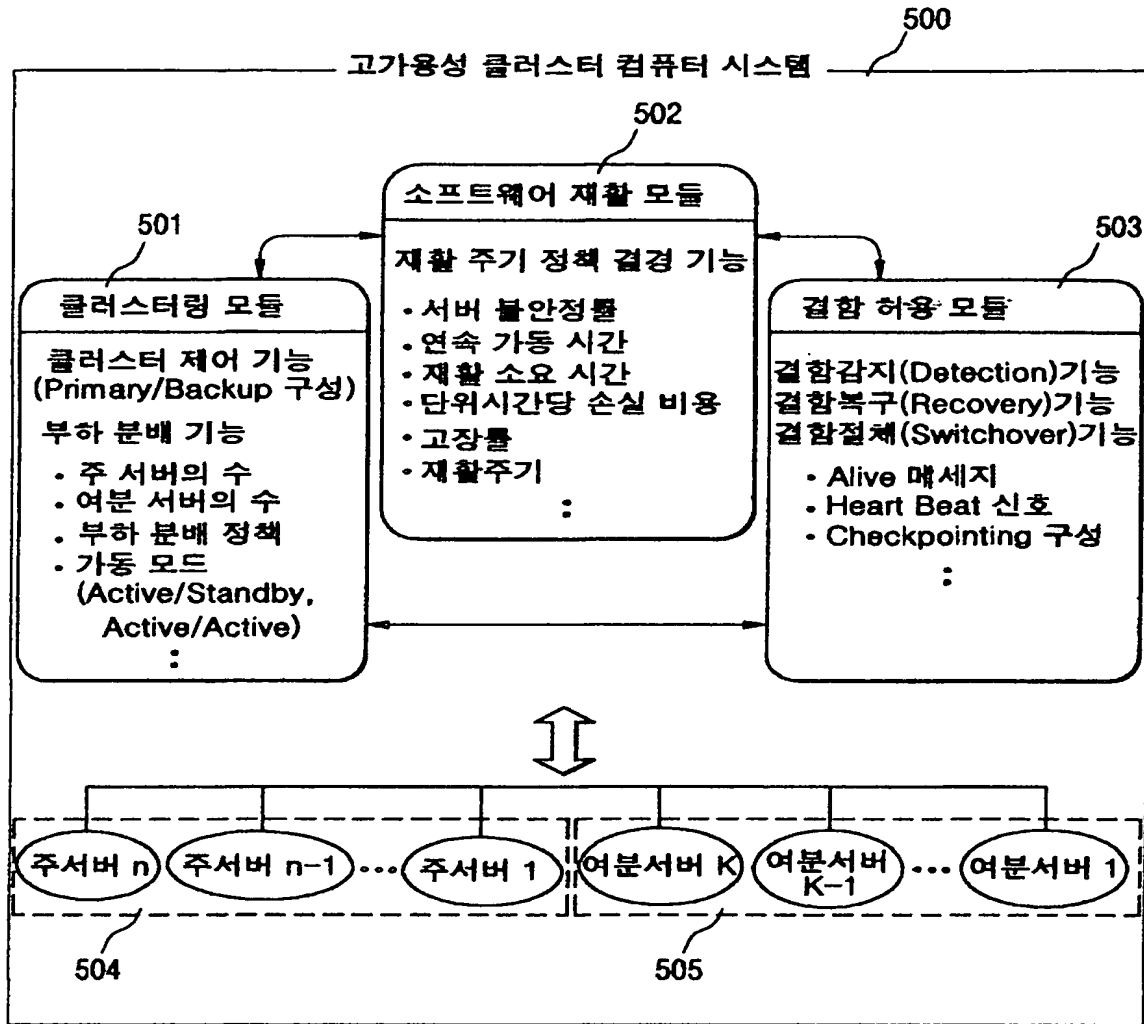
【도 3】



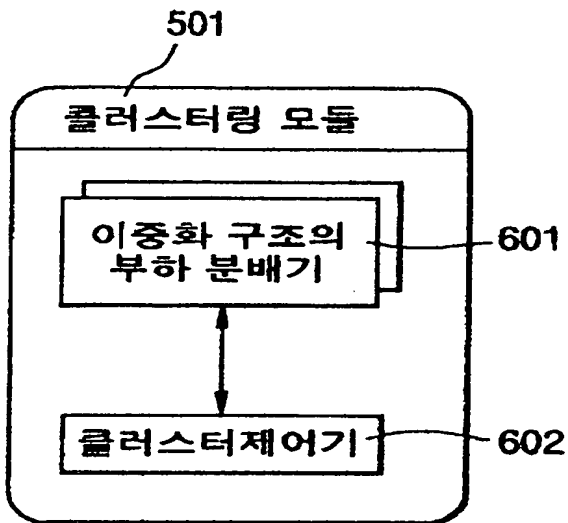
【도 4】



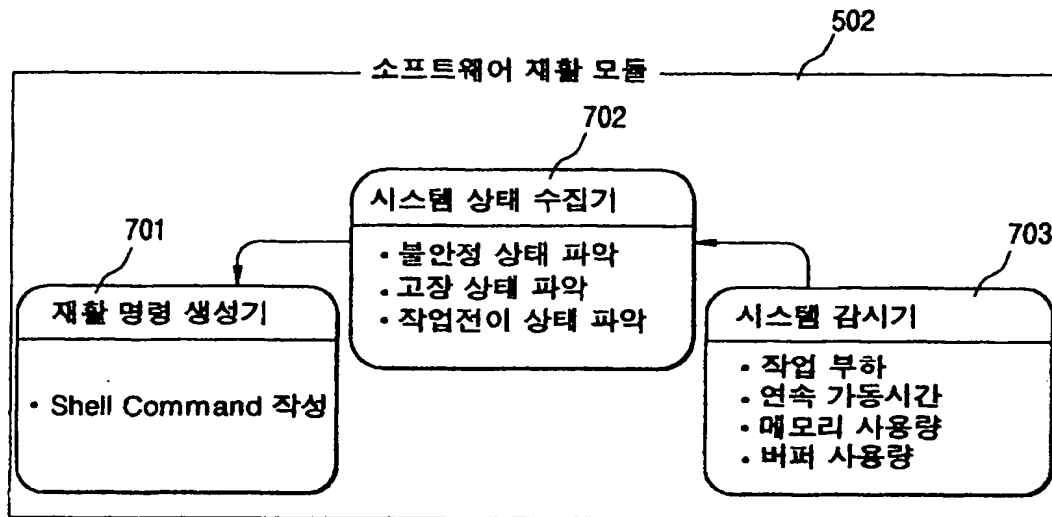
【도 5】



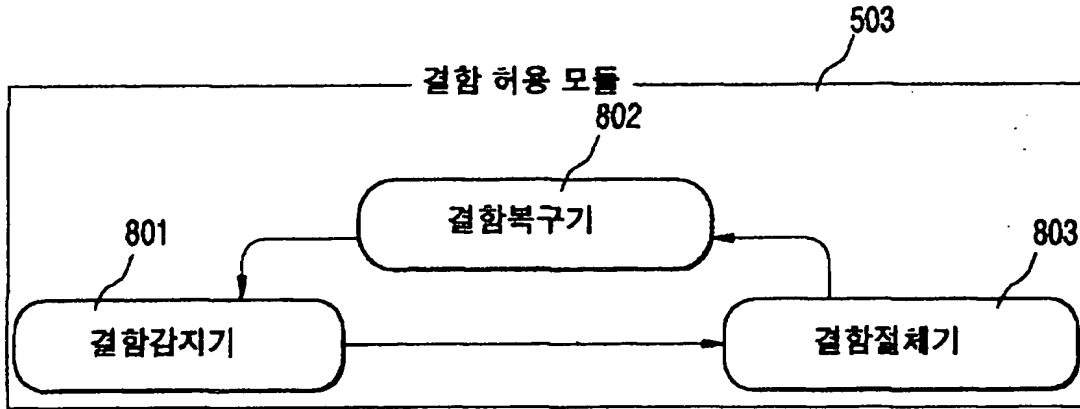
【도 6】



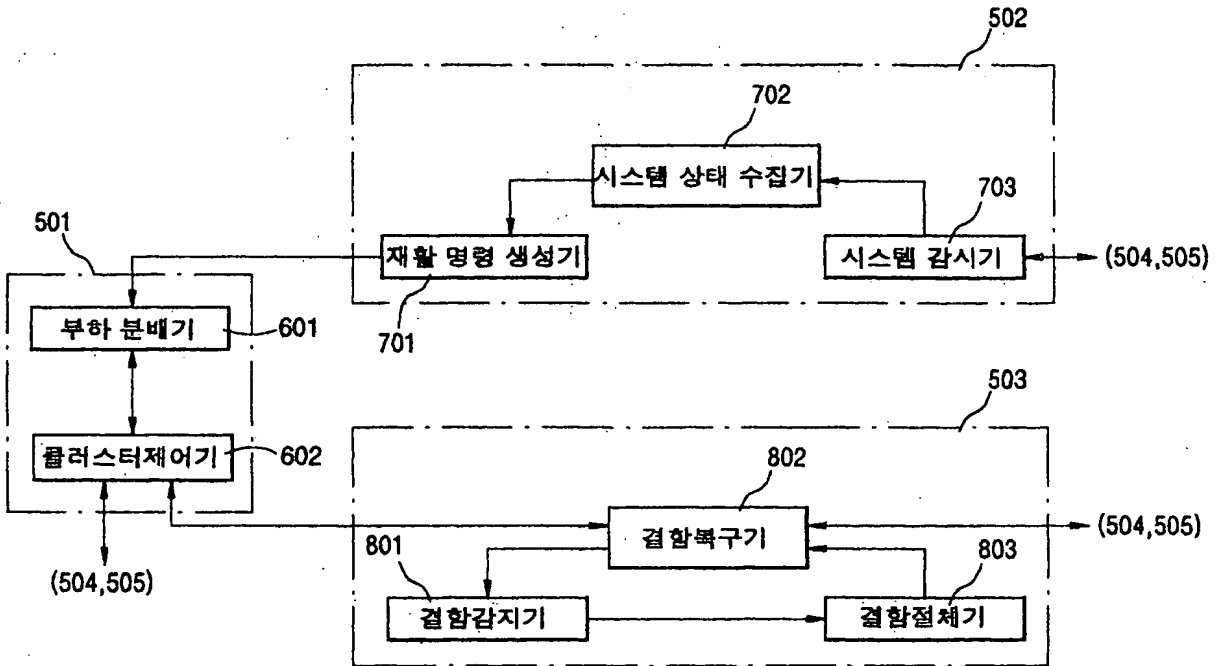
【도 7】



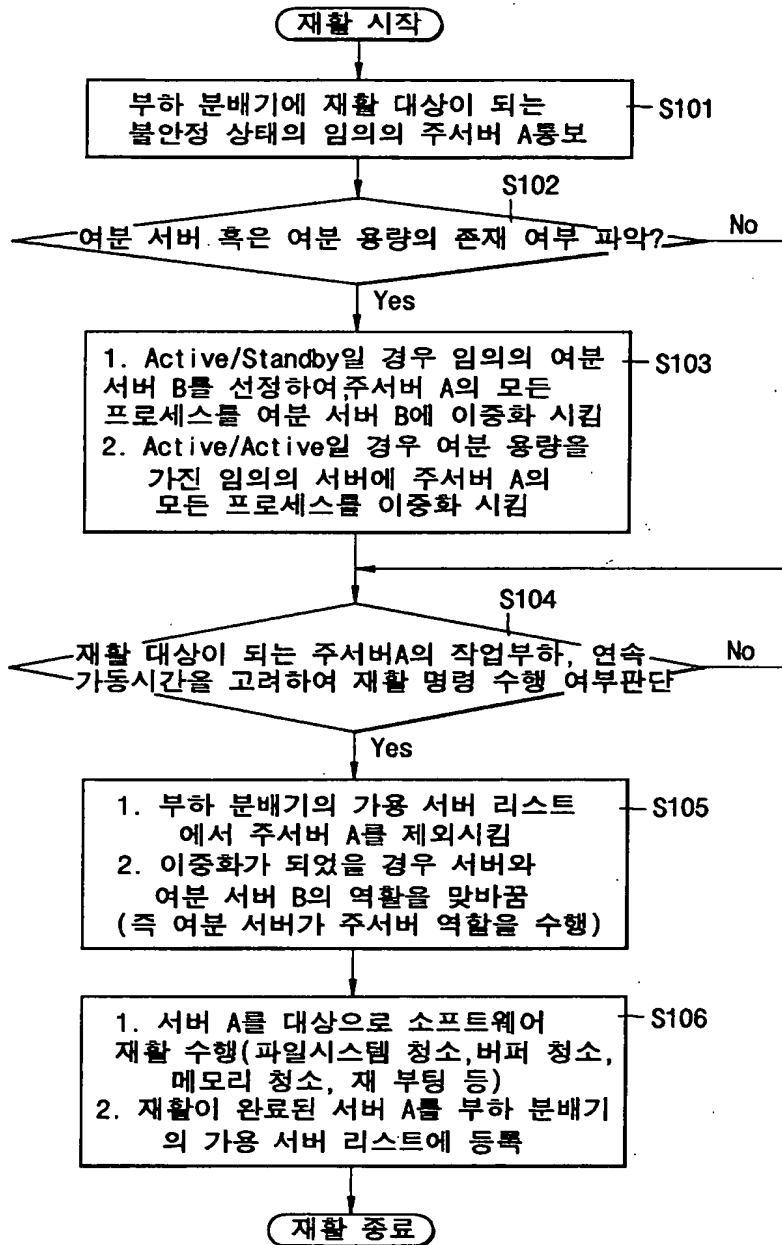
【도 8】



【도 9】



【도 10】



【도 11】

